

ESD-TR-66-80
ESTI FILE COPY

ESD ACCESSION LIST
ESTI Call No. **AL 51729**
Copy No. of cys.
MTR-105

ESD-TR-66-80
ESD RECORD COPY

RETURN TO
SCIENTIFIC & TECHNICAL INFORMATION DIVISION
(ESTI), BUILDING 1211

THE COMPUTATION OF CERTAIN COMMUNICATION CHANNEL
ERROR PROBABILITIES BY AN APPLICATION OF
DIFFERENCE EQUATION METHODS

JULY 1966

S. Berkovits
E. L. Cohen

Prepared for

DEPUTY FOR COMMUNICATIONS SYSTEMS
ELECTRONIC SYSTEMS DIVISION
AIR FORCE SYSTEMS COMMAND
UNITED STATES AIR FORCE
L. G. Hanscom Field, Bedford, Massachusetts



Distribution of this document is unlimited.

Project 7560
Prepared by
THE MITRE CORPORATION
Bedford, Massachusetts
Contract AF19(628)-5165
ADO 636399

ESNO

This document may be reproduced to satisfy official needs of U.S. Government agencies. No other reproduction authorized except with permission of Hq. Electronic Systems Division, ATTN: ESTI.

When US Government drawings, specifications, or other data are used for any purpose other than a definitely related government procurement operation, the government thereby incurs no responsibility nor any obligation whatsoever; and the fact that the government may have formulated, furnished, or in any way supplied the said drawings, specifications, or other data is not to be regarded by implication or otherwise, as in any manner licensing the holder or any other person or corporation, or conveying any rights or permission to manufacture, use, or sell any patented invention that may in any way be related thereto.

Do not return this copy. Retain or destroy.

THE COMPUTATION OF CERTAIN COMMUNICATION CHANNEL
ERROR PROBABILITIES BY AN APPLICATION OF
DIFFERENCE EQUATION METHODS

JULY 1966

S. Berkovits
E. L. Cohen

Prepared for

DEPUTY FOR COMMUNICATIONS SYSTEMS
ELECTRONIC SYSTEMS DIVISION
AIR FORCE SYSTEMS COMMAND
UNITED STATES AIR FORCE
L. G. Hanscom Field, Bedford, Massachusetts



Distribution of this document is unlimited.

Project 7560
Prepared by
THE MITRE CORPORATION
Bedford, Massachusetts
Contract AF19(628)-5165

ABSTRACT

A model for a channel is given. For this model, the recursive method is presented in order to calculate the probability of K symbol errors in a block of n m-bit symbols. The blocks can be interleaved or not.

REVIEW AND APPROVAL

This technical report has been reviewed and is approved.



EDGAR A. GRABHORN, Lt. Colonel, USAF
Director of Communications Development
Deputy for Communications Systems

TABLE OF CONTENTS

	<u>Page</u>
SECTION I THE MODEL	1
SECTION II RECURSIONS - PART I	3
SECTION III RECURSIONS - PART II	6
SECTION IV INPUT FOR THE PROGRAM	10
SECTION V OUTPUT FOR THE PROGRAM	11
APPENDIX PROGRAM	13
REFERENCES	18

SECTION I

THE MODEL

In estimating the performance of an error-correcting device on a specific communication channel, it is necessary to find a meaningful, yet tractable, mathematical model for that channel. An examination of data from real channels suggests that most channels pass through three distinct phases. The first phase, which nearly any error-correcting scheme can handle successfully, is that of long periods of practically error-free transmission. The second phase, which is the antithesis of the first phase in that no scheme can expect to correct it, is that of complete loss of signal for substantial periods of time. The third phase might be described as a generalized sputter or bursts of error bursts, and it is this phase, if it occurs frequently, for which error correctors should be designed. This third phase may sometimes be described by means of a two-state model. The three phases suggest a Markov process with four states, but such a process is mathematically unwieldy. However, when the three phase picture is reasonably correct and the sputter phase is accurately modeled for purposes of estimating coder performance, it is a simple matter to make corrections in such estimates for those periods when transmission is nearly error free or when the signal is lost.

The use of such a two-state model was suggested to us by some work by Gilbert [1]. Gilbert describes a model for binary error distributions

in channels subject to noise bursts. Let $\{x_i\}$ be the error process with $x_i = 1$ for an error in the i^{th} demodulated bit, $x_i = 0$ for no error. Two states, designated G and B, of the channel are postulated such that at the i^{th} bit the state S_i is G or B, and the state S_{i+1} at bit $i + 1$ depends only on S_i . Thus, the state sequence $\{S_i\}$ is a simple Markov chain described by the two transition probabilities, $P: G \rightarrow B$ and $p: B \rightarrow G$. We also use Gilbert's notation $Q = 1 - P$ and $q = 1 - p$ for the $G \rightarrow G$ and $B \rightarrow B$ transitions respectively. Let h, k denote respectively the probabilities of a correct bit in B and in G. Then $P\{x_i=1\} = 1 - h = h'$ if $S_i = B$ and $P\{x_i=1\} = 1 - k = k'$ if $S_i = G$.

Using this model with $k = 1$, Gilbert obtained a good fit to certain phone line error data. The statistic fitted was the probability of occurrence of zero (= error free) runs of length at least K . (The use of $k < 1$ has also been considered by Elliott [5].)

At MITRE, we have found sets of values for the model parameters P, p, h, k , which yield good fits to error data for several different types of communication media. One important statistic involved is the probability of specific error densities in various length blocks.

Given that we have p, P, h, k , [2,3,4], we present the recursive method used to calculate the probability of K symbol errors in a block of n symbols where each symbol consists of m bits.

SECTION II

RECURSIONS - PART I

In Appendix II of [2] (or Appendix B of [3] or [4]) we presented a brief outline of the recursive method used to calculate the probability of K symbol errors in a block of n symbols where each symbol consists of m bits. Now we present the outline in full. Since the last documentation, the technique has been extended to permit the n -symbol blocks to be interleaved or time spread. (To interleave s blocks means to transmit sequentially the first symbol of each of s blocks followed by the second symbol of each of those blocks, etc. Thus, a given m -bit symbol is transmitted s symbol times after the symbol preceding it in its block.)

Letting T and U represent either of the states G and B , we define

$$TOU(t) = P(x_1 = \dots = x_t = 0 \text{ and } S_t = U | S_0 = T)$$

$$TlU(t) = P(\text{for some } i \leq t, x_i = 1 \text{ and } S_t = U | S_0 = T)$$

Then

$$GOG(1) = Qk$$

$$GOB(1) = Ph$$

$$GlG(1) = Qk'$$

$$GlB(1) = Ph'$$

$$BOB(1) = qh$$

$$BOG(1) = pk$$

$$BlB(1) = qh'$$

$$BlG(1) = pk'$$

and

$$GOG(t) = [GOB(t-1) p + GOG(t-1) Q] k$$

$$GlG(t) = [GOB(t-1) p + GOG(t-1) Q] k' + GlB(t-1) p + GlG(t-1) Q$$

$$GOB(t) = [GOB(t-1) q + GOG(t-1) P] h$$

$$G1B(t) = [GOB(t-1) q + GOG(t-1) P] h' + G1B(t-1) q + G1G(t-1) P$$

$$BOB(t) = [BOB(t-1) q + BOG(t-1) P] h$$

$$B1B(t) = [BOB(t-1) q + BOG(t-1) P] h' + B1B(t-1) q + B1G(t-1) P$$

$$BOG(t) = [BOG(t-1) Q + BOB(t-1) p] k$$

$$B1G(t) = [BOG(t-1) Q + BOB(t-1) p] k' + B1B(t-1) p + B1G(t-1) Q$$

Shortly after the program was written, we discovered that $GOG(m)$, $GOB(m)$, $G1G(m)$, $G1B(m)$, $BOG(m)$, $BOB(m)$, $B1G(m)$, $B1B(m)$ could be obtained from a difference equation in powers of J and L (see below). Since on our computer (IBM 7030), it took under a second to compute all eight quantities, we decided not to use the difference equation. However, we work out two, and give all eight results.

$$GOG(t) = \{GOG(t-1) Q + GOB(t-1) p\} k$$

$$GOB(t) = \{GOB(t-1) q + GOG(t-1) P\} h$$

The eigenvalues come from the 2nd order linear difference equation:

$$f_{t+1} - (Qk + qh) f_t - (p-Q) f_{t-1} = 0.$$

that is, $2J = Qk + qh + \sqrt{(Qk + qh)^2 + 4hk(p-Q)}$

and $2L = Qk + qh - \sqrt{(Qk + qh)^2 + 4hk(p-Q)}$

Thus we have

$$GOG(t) = \alpha_1 J^t + \alpha_2 L^t,$$

$$GOB(t) = \beta_1 J^t + \beta_2 L^t$$

and we get $\alpha_1, \alpha_2, \beta_1$ and β_2

from the initial conditions

$$GOG(0) = 1, GOG(1) = Qk$$

and $GOB(0) = 0, GOB(1) = Ph$.

So $\alpha_1 + \alpha_2 = 1, \alpha_1 J + \alpha_2 L = Qk$, which yields

$$GOG(t) = \{(Qk - L)/(J - L)\} J^t + \{(J - Qk)/(J - L)\} L^t$$

Also, $\beta_1 + \beta_2 = 0, \beta_1 J + \beta_2 L = Ph$, which yields

$$GOB(t) = \{Ph/(J - L)\} (J^t - L^t)$$

All eight solutions are as follows:

$$GOG(m) = \{(Qk - L)/(J - L)\} J^m + \{(J - Qk)/(J - L)\} L^m$$

$$GOB(m) = \{Ph/(J - L)\} (J^m - L^m)$$

$$BOG(m) = \{pk/(J - L)\} (J^m - L^m)$$

$$BOB(m) = \{(qh - L)/(J - L)\} J^m + \{(J - qh)/(J - L)\} L^m$$

$$G1G(m) = p/(p + P) + \{P/(p + P)\} (Q-p)^m - \{(J - qh)/(J - L)\} J^m \\ - \{(qh - L)/(J - L)\} L^m$$

$$G1B(m) = \frac{P}{p+P} [1 - (Q-p)^m] - \frac{Ph}{J-L} (J^m - L^m)$$

$$B1B(m) = \frac{P}{p+P} + \frac{P}{p+P} (Q-p)^m - \frac{1}{J-L} [\{J-Qk\} J^m + \{Qk-L\} L^m]$$

$$B1G(m) = \frac{P}{p+P} [1 - (Q-p)^m] - \frac{pk}{J-L} (J^m - L^m)$$

SECTION III

RECURSIONS - PART 2

Again letting T and U represent either of the states G and B,
we define

$$\begin{aligned} \text{TOUI}(s) &= P(x_{(s-1)m+1} = x_{(s-1)m+2} = \dots = x_{sm} = 0, S_{sm} = U | S_0 = T) \\ &= P(m\text{-bit symbol after } s \text{ symbol times is correct and} \\ &\quad \text{ends in state U} | \text{state T}) \end{aligned}$$

$$\begin{aligned} \text{TLUI}(s) &= P(\text{for some } 1 \leq i \leq m, x_{(s-1)m+i} = 1, S_{sm} = U | S_0 = T) \\ &= P(m\text{-bit symbol after } s \text{ symbol times has at least one bit} \\ &\quad \text{error and ends in state U} | \text{state T}) \end{aligned}$$

Let $\text{GXG} = \text{GOG}(m) + \text{G1G}(m)$, $\text{GXB} = \text{GOB}(m) + \text{G1B}(m)$, $\text{BXG} = \text{BOG}(m) + \text{B1G}(m)$,
and $\text{BXB} = \text{BOB}(m) + \text{B1B}(m)$.

Then $\text{GOGI}(s) = \text{GXG} \cdot \text{GOGI}(s-1) + \text{GXB} \cdot \text{BOGI}(s-1)$ and $\text{BOGI}(s) =$
 $\text{BXG} \cdot \text{GOGI}(s-1) + \text{BXB} \cdot \text{BOGI}(s-1)$.

(There will be similar equations in G1BI , B1BI and GOBI , BOBI , and
 G1BI , B1BI , but they have the same eigenvalues and will be omitted.)

$$\text{TOGI}(s) - [\text{GXG} + \text{BXB}] \text{TOGI}(s-1) + [\text{GXG} \cdot \text{BXB} - \text{GXB} \cdot \text{BXG}] \text{TOGI}(s-2) = 0$$

This yields the eigenvalues:

$$2 \sigma = \text{GXG} + \text{BXB} + \sqrt{[\text{GXG} + \text{BXB}]^2 - 4 [\text{GXG} \cdot \text{BXB} - \text{GXB} \cdot \text{BXG}]}$$

$$2 \tau = \text{GXG} + \text{BXB} - \sqrt{[\text{GXG} + \text{BXB}]^2 - 4 [\text{GXG} \cdot \text{BXB} - \text{GXB} \cdot \text{BXG}]}$$

Since $GXG + GXB = BXB + BXG = 1$,

$$\sigma \tau = GXG \cdot BXB - GXB \cdot BXG = (1-GXB) (1-BXG) - GXB \cdot BXG = 1-GXB-BXG$$

$$\sigma + \tau = GXG + BXB = 2 - GXB - BXG = 1 + \sigma \tau .$$

Hence $\sigma = 1$, and $\tau = 1 - GXB - BXG = GXG + BXB - 1$.

Thus we have

$$GOGI(s) = \lambda_1 \cdot 1^s + \lambda_2 \tau^s$$

and $BOGI(s) = \mu_1 \cdot 1^s + \mu_2 \tau^s$

and we get $\lambda_1, \lambda_2, \mu_1$ and μ_2 from the initial conditions.

$$GOGI(1) = GOG(m), GOGI(2) = GXG \cdot GOG(m) + GXB \cdot BOG(m)$$

$$BOGI(1) = BOG(m), BOGI(2) = BXG \cdot GOG(m) + BXB \cdot BOG(m).$$

Hence

$$\lambda_1 \cdot 1 + \lambda_2 \cdot \tau = GOG(m), \lambda_1 \cdot 1^2 + \lambda_2 \cdot \tau^2 = GXG \cdot GOG(m) + GXB \cdot BOG(m).$$

$$\text{Solving, } \lambda_1 = [GOG(m) (1 - BXB) + BOG(m) \cdot GXB] / (1 - \tau)$$

$$\lambda_2 = [GOG(m) BXB - BOG(m) GXG] / (1 - \tau) = [GOG(m) - \lambda_1] / \tau .$$

Also,

$$\mu_1 \cdot 1 + \mu_2 \cdot \tau = BOG(m), \mu_1 \cdot 1^2 + \mu_2 \cdot \tau^2 = BXG \cdot GOG(m) + BXB \cdot BOG(m).$$

$$\text{Solving, } \mu_1 = [BOG(m) (1 - GXG) + BXG \cdot GOG(m)] / (1 - \tau)$$

$$\mu_2 = [BOG(m) - \mu_1] / \tau$$

Therefore, $GOGI(s) = \lambda_1 \cdot 1^s + \lambda_2 \cdot \tau^s$, and $BOGI(s) = \mu_1 \cdot 1^s + \mu_2 \cdot \tau^s$,
 where $\lambda_1, \lambda_2, \mu_1, \mu_2$ are given above. Since s is fixed for any given
 application, we will refer to $GOGI(s)$ as $GOGI$, and to $BOGI(s)$ as $BOGI$.

Consider the first i m -bit symbols of a random interleaved block.
 Let $GB(i,j) = P(j \text{ symbol errors in } i \text{ symbols and } S_{ism} = B | S_0 = G)$.
 Similarly, we define $GG(i,j)$, $BB(i,j)$ and $BG(i,j)$.

Then

$$\begin{array}{ll} GG(1,0) = GOGI & GB(1,0) = GOBI \\ GG(1,1) = G1GI & GB(1,1) = G1BI \\ BG(1,0) = BOGI & BB(1,0) = BOBI \\ BG(1,1) = B1GI & BB(1,1) = B1BI \end{array}$$

Finally, for $i = 2, \dots, n$ and $j = 0, \dots, i$

$$\begin{aligned} GG(i,j) &= GG(i-1,j) GG(1,0) + GB(i-1,j) BG(1,0) \\ &\quad + GG(i-1,j-1) GG(1,1) + GB(i-1,j-1) BG(1,1) \\ GB(i,j) &= GG(i-1,j) GB(1,0) + GB(i-1,j) BB(1,0) \\ &\quad + GG(i-1,j-1) GB(1,1) + GB(i-1,j-1) BB(1,1) \\ BG(i,j) &= BG(i-1,j) GG(1,0) + BB(i-1,j) BG(1,0) \\ &\quad + BG(i-1,j-1) GG(1,1) + BB(i-1,j-1) BG(1,1) \\ BB(i,j) &= BG(i-1,j) GB(1,0) + BB(i-1,j) BB(1,0) \\ &\quad + BG(i-1,j-1) GB(1,1) + BB(i-1,j-1) BB(1,1). \end{aligned}$$

Finally,

$$P(\text{random bit is in G}) = \alpha = \frac{p}{p+P} \quad \text{and hence}$$

$P(K \text{ symbol errors in } n \text{ symbols with } s \text{ blocks interleaved})$

$$= \alpha [GG(n,K) + GB(n,K)] + (1 - \alpha) [BG(n,K) + BB(n,K)]$$

SECTION IV

INPUT FOR THE PROGRAM

XM = No. of bits/symbol

XN = No. of symbols/block

XNEST = Largest number of symbol errors to be considered
(if this field is blank, XNEST = XN)

XIPER = Number of interleaved symbols (0 or 1 means 1)

IK = 0 or not equal to 0 (0 means continue with CP, SP, H, SK;
not equal to 0 means read new parameters)

CP = P

SP = p

H = h

SK = k

SECTION V

OUTPUT OF THE PROGRAM

TIMEX = A8 representation of the time read from IBM 7030 Time Clock
by a STRAP coded routine (one can call his routine or omit
it altogether)

CP, SP, H, SK as above

ALPHA = P (random bit is in G)

M (or XM) as above

N (or XN) as above

NS = No. of recursion terms to be attempted

WPI = P (symbol error)

WY155 = P (no errors when $s = 1$)

WPMU2 = mean number of errors in a block

FCMEAN = mean number of errors given an error occurred (when $s = 1$)

For J, L, A, B see [2,3,4]

IS = number of errors

P = P (IS symbol errors in a block)

R = P (IS symbol errors in a block | error occurred)

Q = P (\leq IS symbol errors in a block)

QH = P ($>$ IS symbol errors in a block)

S = mean number of bits between blocks with $>$ IS symbol errors

PBAR = contribution to mean number of errors per block made by
probabilities actually calculated (if one wants the whole
mean, then NEST = 0 or N; the same applies to VAR, SVAR,
and CMEAN)

VAR = contribution to variance about PBAR made by probabilities
actually calculated by the recursion

SVAR = approximate standard deviation

CMEAN = contribution to mean number of errors given an error
occurred in the block made by probabilities actually
calculated by recursion

APPENDIX

PROGRAM

```

FORTRAN SYSTEM -- VERSION 03/28/65 - CORRECTION LEVEL 03/28/65
C...TO CALCULATE THE RECURSION PROBABILITIES
00000 COMMON 777/ TIMEX
00001 DIMENSION GG(515), GB(515), BG(515), BB(515)
00002 DIMENSION B0B(205), B1B(205), B0G(205), B1G(205)
00003 DIMENSION G0G(205), G1G(205), G0B(205), G1B(205)
00004 INTEGER XM(50), XN(50), XNES(50), XIPER(50)
00005 404 READ 701, LK
00006 1701 FORMAT (12)
C...READ IN THE NO. OF BITS PER SYMBOL, THE NO. OF SYMBOLS,
C HOW MANY SYMBOLS TO GO THRU, HOW MANY SYMBOLS TO INTERLEAVE
00007 READ 701, (XM(L), XN(L), XNES(L), XIPER(L), L=1,LK)
00008 701 FORMAT (13, 1X, 16, 2X, 13, 1X, 14)
C...READ IN CP, SP, H, SK
00009 604 READ 702, 1K, CP, SP, H, SK
00010 702 FORMAT (11, E17.10, 3(E18.11))
00011 DO 2000 K=1,LK
00012 M = XM(K)
00013 N = XN(K)
00014 NEST = XNES(K)
00015 IPERD = XIPER(K)
00016 IF(NEST.EQ.0) NEST = N
00017 NS = NEST + 1
00018 CALL TIME
00019 PRINT 300, TIMEX
00020 300 FORMAT(1H1,A8,21X, 77H*CALCULATION OF GILBERT CHANNEL ERROR PROBAB
ILITIES USING RECURSION FORMULAS*//////)
00021 MUD = M*N
00022 FMUD = FLOAT(MUD)
00023 MUD1 = MUD - 1
00024 HCOMP = 1. - H
00025 SKCOMP = 1. - SK
00026 CQ = 1. - CP
00027 SQ = 1. - SP
00028 GAMMA = 1./(CP+SP)
00029 ALPHA = SP*GAMMA
00030 BETA = CP*GAMMA
00031 P1 = (SP*SKCOMP + CP*HCOMP)*GAMMA
00032 PRINT 301, CP, SP, H, SK, ALPHA
00033 301 FORMAT (5X, 27HCHANNEL PARAMETERS CP = ,E11.6, 5X, 5HSP = ,
E11.6, 5X, 5H H = , F8.6, 5X, 5H K = ,E16.10, 5X, 8HALPHA = , F8.6)
00034 PRINT 302, M, N, NS, P1, IPERD
00035 302 FORMAT (8X, 24HCODE PARAMETERS M = , 14, 13X, 4HN = , 15,
112X, 35HNO. OF RECURSION TERMS REQUESTED = , 15/ 7X, $P1 = $,
2 E13.7, 75X, 14, $ BLOCKS INTERLEAVED
3 $,/)

```

```

00036      WN = SP - CQ
00037      WT = SK*CQ + H*SQ
00038      WR = WT**2 + 4.*H*SK*WN
00039      WSQRT = SQRT(WR)
00040      WJ = (WT + WSQRT)/2.
00041      WL = (WT - WSQRT)/2.
00042      WA = WJ + WN * (CP*SK*HCUMP + SP*H*SKCOMP) /
1      (CP*HCUMP + SP*SKCOMP)
00043      WA = WA / WSQRT
00044      WB = WA - 1.
00045      WX2 = (WA*(1.-WJ**M)/(1.-WJ)) - (WB*(1.-WL**M)/(1.-WL))
00046      WPI = P1 * WX2
00047      WPMU2 = FLOAT(N) * WPI
00048      KJ = FLOAT(MUD) * ALOG10(WJ)

```

```

00049      RL = FLOAT(MUD) * ALOG10(WL)
00050      IF(RJ. LT. -50.) F = 0.
00051      IF(RJ. GE. -50.) F = WJ**MUD
00052      IF(RL. LT. -50.) G = 0.
00053      IF(RL. GE. -50.) G = WL**MUD
00054      WY155 = 1. - ((WA*(1.-F)/(1.-WJ)) - WB*(1.-G)/(1.-WL))*P1
00055      FCMEAN = WPMU2 / (1. - WY155)
00056      PRINT 806, WPI, WY155, WPMU2, FCMEAN, WJ, WL, WA, WB
00057      806 FORMAT (1X, 20HPRB SYMBOL ERROR = ,E12.6, 1H*, 2X, 17HPRB NO ER
1KORS = ,E12.6, 1H*, 2X, 8H MEAN = ,E12.6, 1H*, 2X, 30HCONDITIONAL ME
2AN = ,E12.6, 1H*, 2X, 10X, 4HJ = , 3X, E16.10, 1H*,
315X, 4HL = , 3X, F9.6, 1H*, 6X, 4HA = , 3X, F9.6, 1H*, 20X, 4HB =
4,F9.6, 1H*///)
00058      GOG(1) = CQ*SK
00059      GIG(1) = CQ - GOG(1)
00060      GOB(1) = CP*H
00061      GIB(1) = CP - GOB(1)
00062      BOB(1) = SQ*H
00063      BIB(1) = SQ - BOB(1)
00064      BOG(1) = SP*SK
00065      BIG(1) = SP - BOG(1)
00066      IF(M.EQ.1) GO TO 80
00067      DO 50 J=2,M
00068      J1 = J-1
00069      TA = GOB(J1)*SP + GOG(J1)*CQ
00070      GOG(J) = TA*SK
00071      GIG(J) = TA - GOG(J) + GIG(J1)*CQ + GIB(J1)*SP
00072      TB = GOB(J1)*SQ + GOG(J1)*CP
00073      GOB(J) = TB*H
00074      GIB(J) = TB - GOB(J) + GIB(J1)*SQ + GIG(J1)*CP
00075      TC = BOB(J1)*SQ + BOG(J1)*CP
00076      BOB(J) = TC*H
00077      BIB(J) = TC - BOB(J) + BIB(J1)*SQ + BIG(J1)*CP
00078      TD = BOG(J1)*CQ + BOB(J1)*SP
00079      BOG(J) = TD*SK
00080      50 BIG(J) = TD - BOG(J) + BIG(J1)*CQ + BIB(J1)*SP
00081      80 PRINT 303, GOG(M), GIG(M), GOB(M), GIB(M)

```

```

00082      303 FORMAT (5X, 6MGOG = , E12.5, 5X, 6MGIG = , E12.5, 5X, 6MGOB = ,
      1E12.5, 5X, 6MGIB = , E12.5)
00083      PRINT 304, BOG(M), BIG(M), BOB(M), BIB(M)
00084      304 FORMAT (5X, 6HBOG = , E12.5, 5X, 6HBIG = , E12.5, 5X, 6HBOB = ,
      1E12.5, 5X, 6HBIB = , E12.5//)
00085      RPI1 = ALPHA*(GIB(M) + GIG(M)) + BETA*(BIB(M) + BIG(M))
00086      PRINT 505, RPI1
00087      505 FORMAT (1X, 30HPROB RANDOM SYMBOL IN ERROR = , E12.6, 1H*///)
00088      CALL TIME
00089      GG(1) = GOG(M)
00090      GG(2) = GIG(M)
00091      GB(1) = GOB(M)
00092      GB(2) = GIB(M)
00093      BG(1) = BOG(M)
00094      BG(2) = BIG(M)
00095      BB(1) = BOB(M)
00096      BB(2) = BIB(M)
00097      IF (IPERD .NE. 0) GO TO 9001
00098      GOGI = GOG(M)
00099      GIGI = GIG(M)
00100      GOBI = GOB(M)
00101      GIBI = GIB(M)
00102      BOGI = BOG(M)
00103      BIGI = BIG(M)

```

```

00104      BOBI = BOB(M)
00105      BIBI = BIB(M)
00106      GO TO 9002
00107      9001 GXG = GG(1) + GG(2)
00108      GXB = GB(1) + GB(2)
00109      BXG = BG(1) + BG(2)
00110      BXB = BB(1) + BB(2)
00111      SIG = 1.
00112      TAU = GXG + BXB - 1.
00113      DEN = 1. - TAU
00114      GT = 1. - BXB
00115      BT = 1. - GXG
00116      G = 0.
00117      PERD = FLOAT(IPERD)
00118      IF (PERD * ALOG10(TAU) .GE. -300.) G = TAU ** PERD
00119      PRINT 9003, SIG, TAU
00120      9003 FORMAT (8X, $SIGMA = $, E20.12, 10X, $TAU = $, E20.12/'
00121      AG = (GG(1) * GT + GXB * BG(1)) / DEN
00122      CG = (GG(1) - AG) / TAU
00123      AB = (BXG * GG(1) + BG(1) * BT) / DEN
00124      CB = (BG(1) - AB) / TAU
00125      GOGI = AG + CG * G
00126      BOGI = AB + CB * G
00127      AG = (GG(2) * GT + GXB * BG(2)) / DEN
00128      CG = (GG(2) - AG) / TAU
00129      AB = (BXG * GG(2) + BG(2) * BT) / DEN

```

```

00130      CB = (BG(2) - AB) / TAU
00131      G1G1 = AG      + CG * G
00132      B1G1 = AB      + CB * G
00133      AG = (GB(1) * GT + GXB * BB(1)) / OEN
00134      CG = (GB(1) - AG) / TAU
00135      AB = (BXG * GB(1) + BB(1) * BT) / DEN
00136      CB = (BB(1) - AB) / TAU
00137      GCB1 = AG      + CG * G
00138      BOB1 = AB      + CB * G
00139      AG = (GB(2) * GT + GXB * BB(2)) / OEN
00140      CG = (GB(2) - AG) / TAU
00141      AB = (BXG * GB(2) + BB(2) * BT) / DEN
00142      CB = (BB(2) - AB) / TAU
00143      G1B1 = AG      + CG * G
00144      B1B1 = AB      + CB * G
00145      PRINT 9005,GOG1, G1G1, GOB1, G1B1, BOG1, B1G1, BOB1, B1B1
00146      9005 FORMAT (7X, 7HGOG1 = ,E12.5, 4X, 7HG1G1 = , E12.5, 4X, 7HGOB1 = ,
1 E12.5, 4X, 7HG1B1 = , E12.5/ 5X, 7HBOG1 = , E12.5, 4X, 7HB1G1 = ,
2 E12.5, 4X, 7HBOB1 = , E12.5, 4X, 7HB1B1 = , E12.5//)
00147      9002 IF(NEST.EQ.1) GO TO 90
C...CALCULATE THE REKUSION PROBABILITIES
00148      DO 51 L1=2,N
00149      LZ = MIN0(L1,NEST)
00150      LY = L1
00151      IF(L1 .GT. NEST) LY = NEST + 1
00152      IF(L1 .GT. NEST) GO TO 97
00153      N1 = L1 + 1
00154      GG(N1) = GG(L1)*G1G1 + GB(L1)*B1G1
00155      GB(N1) = GG(L1)*G1B1 + GB(L1)*B1B1
00156      BG(N1) = BG(L1)*G1G1 + BB(L1)*B1G1
00157      BB(N1) = BG(L1)*G1B1 + BB(L1)*B1B1
00158      97 CONTINUE
00159      DO 52 L2=2,LY
00160      N2 = LY - L2 + 2
00161      N3 = N2 - 1
00162      YAH1 = GG(N2)*GOG1 + GG(N3)*G1G1 + GB(N2)*BOG1 + GB(N3)*B1G1
00163      GB(N2) = GB(N2)*BOB1 + GB(N3)*B1B1 + GG(N2)*GOB1 + GG(N3)*G1B1
00164      GG(N2) = YAH1
00165      YAH2 = BG(N2)*GOG1 + BG(N3)*G1G1 + BB(N2)*BOG1 + BB(N3)*B1G1
00166      BB(N2) = BB(N2)*BOB1 + BB(N3)*B1B1 + BG(N2)*GOB1 + BG(N3)*G1B1
00167      52 BG(N2) = YAH2
00168      YAH3 = GG(1)*GOG1 + GB(1)*BOG1
00169      GB(1) = GB(1)*BOB1 + GG(1)*GOB1
00170      GG(1) = YAH3
00171      YAH4 = BG(1)*GOG1 + BB(1)*BOG1
00172      BB(1) = BB(1)*BOB1 + BG(1)*GOB1
00173      BG(1) = YAH4
00174      51 CONTINUE
00175      90 CONTINUE
00176      IF(NEST.GT.35) PRINT 503

```

```

00177 503 FORMAT(//////////50X, 33H(SYMBOL DATA STARTS ON NEXT PAGE))
00178 IF(NEST.GT.35) PRINT 502
00179 502 FORMAT (1H1)
00180 PRINT 500, TIMEX
00181 500 FORMAT(1X,A8/ 1X, $SYMBOL NUMBER$10X, 15HRECURSION PROBS, 8X,
1 17HCONDITIONAL PROBS, 15X, 13HCUM RECURSION, 12X, $1. - CUM$,
2 9X, $MEAN 'TWEENS')
00182 P = ALPHA*(GG(1) + GB(1)) + BETA *(BG(1) + BB(1))
00183 Q = P
00184 R1 = 1. - P
00185 S1 = Q * FMUD / R1
00186 PRINT 507, P, Q, R1, S1
00187 507 FORMAT (12X,$0*$, 8X, E16.9, 1H*, 34X, E18.12, 1H*, 9X,
1 E10.4, $*$, 9X, E10.4, $*$)
00188 PBAR = 0.
00189 PVAR = 0.
00190 GO 53 I=2,NS
00191 IS = 1 - 1
00192 P = ALPHA*(GG(1) + GB(1)) + BETA *(BG(1) + BB(1))
00193 Q = P + Q
00194 QH = 1. - Q
00195 R = P / R1
00196 S = Q * FMUD / QH
00197 FIS = FLOAT(IS)
00198 FIST= FIS**2
00199 PBAR = PBAR + P * FIS
00200 PVAR = PVAR + P * FIST
00201 PRINT 501, IS, P, R, Q, QH, S
00202 53 CONTINUE
00203 501 FORMAT ( 8X, 15, 1H*, 2( 9X, E15.8, 1H*), 9X, E18.12, 1H*, 9X,
1 E10.4, 1H*, 9X, E10.4, $*$)
00204 606 CONTINUE
00205 VAR = PVAR - PBAR*PBAR
00206 SVAR = SQRT(VAR)
00207 CMEAN = PBAR / R1
00208 CALL TIME
00209 PRINT 305, PBAR, VAR, SVAR, CMEAN, TIMEX
00210 305 FORMAT (// 25X, $PARTIAL MEAN = $, E13.6/
1 25X, $APPROXIMATE VARIANCE = $, E13.6/
2 25X, $APPROXIMATE ST. DEV. = $, E13.6/
3 25X, $PARTIAL CONDITIONAL MEAN = $, E13.6//1X, A8//)
00211 2000 CONTINUE
00212 IF(IK .EQ. 0) GO TO 604
00213 GO TO 404
00214 END

```

REFERENCES

1. E.N. Gilbert, Capacity of a Burst-Noise Channel, Bell System Tech. J., 39, Sept. 1960.
- *2. S. Berkovits, E. L. Cohen, and N. Zierler, A Model for Digital Error Distributions, The MITRE Corp., Bedford, Mass., ESD-TR-65-146, 1965.
- *3. S. Berkovits, E.L. Cohen, and N. Zierler, A Model for Digital Error Distributions, The MITRE Corp., Bedford, Mass., TM-4189, March 15, 1965.
- *4. S. Berkovits, E. L. Cohen, and N. Zierler, A Model for Digital Error Distributions, First IEEE Ann. Commun. Conv., Boulder, Colo., June 7-9, 1965.
5. E. O. Elliott, Estimates of Error Rates for Codes on Burst-Noise Channels, Bell System Tech. J., 42, Sept. 1963.

* References 2,3,4 are essentially the same, 2 is more accurate than 3, and 3 is more accurate than 4.

DOCUMENT CONTROL DATA - R&D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY (Corporate author) The MITRE Corporation Bedford, Massachusetts		2a. REPORT SECURITY CLASSIFICATION Unclassified	
		2b. GROUP	
3. REPORT TITLE The Computation of Certain Communication Channel Error Probabilities by an Application of Difference Equation Methods			
4. DESCRIPTIVE NOTES (Type of report and inclusive dates) N/A			
5. AUTHOR(S) (Last name, first name, initial) Berkovits, Shimson, and Cohen, Edward L.			
6. REPORT DATE July 1966		7a. TOTAL NO. OF PAGES 21	7b. NO. OF REFS 5
8a. CONTRACT OR GRANT NO. AF19(628)-5165		9a. ORIGINATOR'S REPORT NUMBER(S) ESD-TR-66-80	
b. PROJECT NO. 7560		9b. OTHER REPORT NO(S) (Any other numbers that may be assigned this report) MTR-105	
c.			
d.			
10. AVAILABILITY/LIMITATION NOTICES Distribution of this document is unlimited.			
11. SUPPLEMENTARY NOTES		12. SPONSORING MILITARY ACTIVITY Deputy for Communications Systems, Electronic Systems Division, L.G. Hansom Field, Bedford, Mass.	
13. ABSTRACT A model for a channel is given. For this model, the recursive method is presented in order to calculate the probability of K symbol errors in a block of n m-bit symbols. The blocks can be interleaved or not.			

14	KEY WORDS	LINK A		LINK B		LINK C	
		ROLE	WT	ROLE	WT	ROLE	WT
Channel Modelling Communications Error Control							

INSTRUCTIONS

1. ORIGINATING ACTIVITY: Enter the name and address of the contractor, subcontractor, grantee, Department of Defense activity or other organization (corporate author) issuing the report.

2a. REPORT SECURITY CLASSIFICATION: Enter the overall security classification of the report. Indicate whether "Restricted Data" is included. Marking is to be in accordance with appropriate security regulations.

2b. GROUP: Automatic downgrading is specified in DoD Directive 5200.10 and Armed Forces Industrial Manual. Enter the group number. Also, when applicable, show that optional markings have been used for Group 3 and Group 4 as authorized.

3. REPORT TITLE: Enter the complete report title in all capital letters. Titles in all cases should be unclassified. If a meaningful title cannot be selected without classification, show title classification in all capitals in parentheses immediately following the title.

4. DESCRIPTIVE NOTES: If appropriate, enter the type of report, e.g., interim, progress, summary, annual, or final. Give the inclusive dates when a specific reporting period is covered.

5. AUTHOR(S): Enter the name(s) of author(s) as shown on or in the report. Enter last name, first name, middle initial. If military, show rank and branch of service. The name of the principal author is an absolute minimum requirement.

6. REPORT DATE: Enter the date of the report as day, month, year; or month, year. If more than one date appears on the report, use date of publication.

7a. TOTAL NUMBER OF PAGES: The total page count should follow normal pagination procedures, i.e., enter the number of pages containing information.

7b. NUMBER OF REFERENCES: Enter the total number of references cited in the report.

8a. CONTRACT OR GRANT NUMBER: If appropriate, enter the applicable number of the contract or grant under which the report was written.

8b, 8c, & 8d. PROJECT NUMBER: Enter the appropriate military department identification, such as project number, subproject number, system numbers, task number, etc.

9a. ORIGINATOR'S REPORT NUMBER(S): Enter the official report number by which the document will be identified and controlled by the originating activity. This number must be unique to this report.

9b. OTHER REPORT NUMBER(S): If the report has been assigned any other report numbers (either by the originator or by the sponsor), also enter this number(s).

10. AVAILABILITY/LIMITATION NOTICES: Enter any limitations on further dissemination of the report, other than those

imposed by security classification, using standard statements such as:

- (1) "Qualified requesters may obtain copies of this report from DDC."
- (2) "Foreign announcement and dissemination of this report by DDC is not authorized."
- (3) "U. S. Government agencies may obtain copies of this report directly from DDC. Other qualified DDC users shall request through _____."
- (4) "U. S. military agencies may obtain copies of this report directly from DDC. Other qualified users shall request through _____."
- (5) "All distribution of this report is controlled. Qualified DDC users shall request through _____."

If the report has been furnished to the Office of Technical Services, Department of Commerce, for sale to the public, indicate this fact and enter the price, if known.

11. SUPPLEMENTARY NOTES: Use for additional explanatory notes.

12. SPONSORING MILITARY ACTIVITY: Enter the name of the departmental project office or laboratory sponsoring (paying for) the research and development. Include address.

13. ABSTRACT: Enter an abstract giving a brief and factual summary of the document indicative of the report, even though it may also appear elsewhere in the body of the technical report. If additional space is required, a continuation sheet shall be attached.

It is highly desirable that the abstract of classified reports be unclassified. Each paragraph of the abstract shall end with an indication of the military security classification of the information in the paragraph, represented as (TS), (S), (C), or (U).

There is no limitation on the length of the abstract. However, the suggested length is from 150 to 225 words.

14. KEY WORDS: Key words are technically meaningful terms or short phrases that characterize a report and may be used as index entries for cataloging the report. Key words must be selected so that no security classification is required. Identifiers, such as equipment model designation, trade name, military project code name, geographic location, may be used as key words but will be followed by an indication of technical context. The assignment of links, rules, and weights is optional.